

Time-Contrastive Learning Based Unsupervised DNN Feature Extraction for Speaker Verification

Achintya Kr. Sarkar and Zheng-Hua Tan

Department of Electronic Systems, Aalborg University, Denmark

akc@es.aau.dk, zt@es.aau.dk

Abstract

In this paper, we present a time-contrastive learning (TCL) based unsupervised bottleneck (BN) feature extraction method for speech signals with an application to speaker verification. The method exploits the temporal structure of a speech signal and more specifically, it trains deep neural networks (DNNs) to discriminate temporal events obtained by uniformly segmenting the signal without using any label information, in contrast to conventional DNN based BN feature extraction methods that train DNNs using labeled data to discriminate speakers or pass-phrases or phones or a combination of them. We consider different strategies for TCL and its combination with transfer learning. Experimental results on the RSR2015 database show that the TCL method is superior to the conventional speaker and pass-phrase discriminant BN feature and Mel-frequency cepstral coefficients (MFCCs) feature for text-dependent speaker verification. The unsupervised TCL method further has the advantage of being able to leverage the huge amount of unlabeled data that are often available in real life.

Index Terms: Unsupervised time-contrastive learning, DNNs, Bottleneck feature, Speaker verification

1. Introduction

Speaker verification is the task of either accepting or rejecting a person by his/her voice. It is broadly divided into two sub-categories: text-dependent (TD) and text-independent (TI) speaker verification (SV). In TD-SV, users are constrained to speak the same pass-phrase during both enrollment and test phases, whereas in TI-SV, speakers are free to speak any sentences (of text) during training and test phases. Since TD-SV relies on the same sentence, i.e. a matched phonetic content, during both training and test phases, it outperforms the TI-SV counterpart for short speech utterances.

Speech is a quasi stationary signal and hence short-time based cepstral feature representations are mostly used in speaker [1] and speech recognition [2] systems. Recently, the deep neural network (DNN) concept [3] has gained great interest in the field of automatic speech and speaker recognition as DNNs are capable of modeling highly nonlinear structure of patterns in data. In the context of speaker verification, DNNs are found to be used either for extracting posteriori statistics [4, 5, 6] with respect to the pre-defined phonetic classes for i-vector extraction or for discriminative feature extraction, where a multi-layer DNN is trained with objective to discriminate speakers, pass-phrases, phonetic classes, phones or a combination of them. In case of i-vector extraction, a speech utterance is aligned against

a DNN based automatic speech recognition (ASR) for sufficient statistics with respect to a pre-defined phonetic classes (called senones) to incorporate the phonetic knowledge in i-vectors. In case of feature extraction, the outputs of the DNN hidden layers are used to vectorize characterization of speech data, which are either directly used for speaker characterization *called d-vector* [7] analogous to the i-vector concept or projected onto a low dimensional space *called bottleneck (BN) feature* [8, 9, 10, 11] for speaker recognition. It has been demonstrated in [9, 10, 11] that DNN based systems either perform better than or provide complementary information to conventional short-time cepstral feature based speaker recognition systems.

In conventional BN feature extraction methods, DNNs are generally optimized to discriminate speakers or pass-phrases or phones or a combination of them in the training data. It is observed in [9] that the performances of TD-SV systems using BN features trained on discriminating speaker+pass-phrases or phonetic or speaker+phonetic are similar to each other. However, their performances are better than that of a cepstral feature. All of these DNN BN feature extraction methods exploit the labeling/supervision information of data. The success of these systems highly depend on obtaining good labeled data. They can be categorized as supervised or semi-supervised approaches. Unsupervised learning methods, although being challenging to deploy, are more attractive as they are able to take advantage of huge amount of unlabeled data available in real-life.

Inspired by the recently proposed time contrastive learning (TCL) algorithm for classification of EEG/MPEG data in [12], in this paper, we explore the TCL concept for speech signals aiming at finding out whether it is applicable for speech and what are the effective strategies. We present an *unsupervised* bottleneck feature extraction method for speech, which focuses on discriminating the temporal events across the speech signal with no need for any speaker or pass-phrase or phonetic labels in contrast to the conventional DNN based BN feature extraction approaches that discriminate speakers/pass-phrases/phones/both during training using labeled data. The main idea is to exploit the *temporal non-stationary* structure in the speech signal and discriminate the temporal events in an *unsupervised manner*. Specifically, speech utterances are first *uniformly segmented* into, e.g. N segments, and data-points (frames) within a particular segment are then considered belonging to a single class. N segments therefore constitute N classes. Next, a DNN is trained to discriminate the data of the different segments. Finally, the output of the DNN hidden layer is projected onto a low dimensional space to get BN features for speaker verification. Since we do not consider any speaker/pass-phrase/phonetic label/alignment information during the segmentation of speech utterances in the TCL approach, we call it an *unsupervised DNN BN feature*. Furthermore, We consider different strategies for TCL and its combination with transfer learning.

The paper reflects some results from the OCTAVE Project (#647850), funded by the Research European Agency (REA) of the European Commission, in its framework programme Horizon 2020. The views expressed in this paper are those of the authors and do not engage any official position of the European Commission.

We compare the performance of the unsupervised TCL based BN feature with those of cepstral and conventional BN features in text-dependent speaker verification on RSR2015 consisting of short utterances. We show that the proposed BN feature gives better performance for TD-SV than cepstral and conventional BN features. Gaussian mixture model - universal background model (GMM-UBM) [13] technique is used for speaker verification, since it is well established that a GMM based classifier [14, 15] gives better performance in SV using short utterances than the i-vector [16].

2. Conventional DNN bottleneck features

Among conventional DNN bottleneck features, the BN feature in [9] is based on DNNs trained to optimize two cross-entropy based objective functions simultaneously: one for discriminating speakers and the other for discriminating pass-phrases. The output of the last DNN hidden layer is connected to two types of cross entropy nodes: one predicting speakers and the other predicting pass-phrases. The equally weighted combination of these two criteria is used as a final criterion and DNN multi-task learning procedure is followed [17]. The output of a DNN hidden layer is used frame-level features called deep features. *Bottleneck features* are then obtained by projecting the high dimensional deep feature vectors onto a lower dimensional space. Fig.1 illustrates the conventional DNN based BN feature extraction method. It is observed in [9] that the performance of text-dependent speaker verification using BN features extracted based on the discrimination of both speaker and pass-phrases is similar to that of features based on discrimination of either speakers or speaker+phone. Among them, augmentation of a cepstral feature with the speaker+pass-phrase discriminant BN feature yields the lowest error rates. In this work, we consider speaker+pass-phrase discrimination and different hidden layers of the DNN for the conventional bottleneck feature extraction.

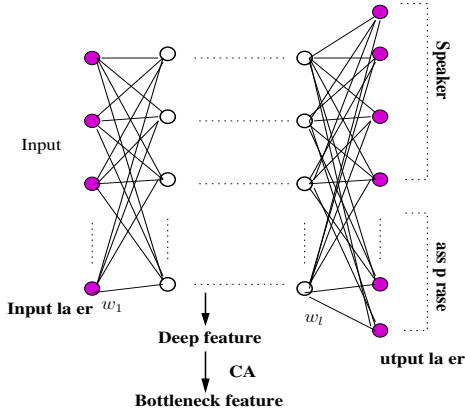


Figure 1: Conventional DNN speaker + pass-phrase based discriminated bottleneck feature extraction.

3. Time-contrastive learning

In the TCL concept [12], multivariate time series data X are first divided into a number of uniform segments (say N), and then *all data-points* within a particular segment are assigned to one class label as follows:

$$\underbrace{(x_1, x_M), \dots, (x_{iM+1}, x_{iM+M}), \dots, (x_{(N-1)M+1}, x_{NM})}_{\text{class 1}} \dots \underbrace{\dots}_{\text{class } N}$$

where i and M indicate the segment index and the number of data points within a segment for a particular data series, respectively. Finally, a DNN is trained to classify the data across the segments. The output of the last hidden layer is used as a feature. The method is evaluated on brain imaging data, specifically magnetoencephalography (MEG) signals, to classify the different states of brain and the task involves a classification of few number of classes (only four classes) [12]. Due to the significant difference between speech and MEG signals in nature, this work explores the potential of the TCL concept for speech feature extraction and its application to speaker verification as an example application.

4. Unsupervised DNN features

Inspired by the TCL algorithm [12], we explore the concept on speech to generate bottleneck feature for speaker verification. Since the patterns varying across the speech signal highly depend on the contents of spoken words, we consider two TCL training approaches and their combinations with transfer learning.

4.1. TCL learning

Similarly to the TCL algorithm [12], speech utterances are uniformly segmented into a number of *pre-defined* segments (say N) regardless of speakers and contents, and data within a particular segment are assigned one class label distinct from the other segments. A DNN is then trained by pooling together all training utterances of different text contents and different speakers, to optimize the cross-entropy objective function for classifying the data across the different segments, i.e. discriminating the temporal events of the speech signal in *unsupervised manner*. Fig.2 illustrates the method.

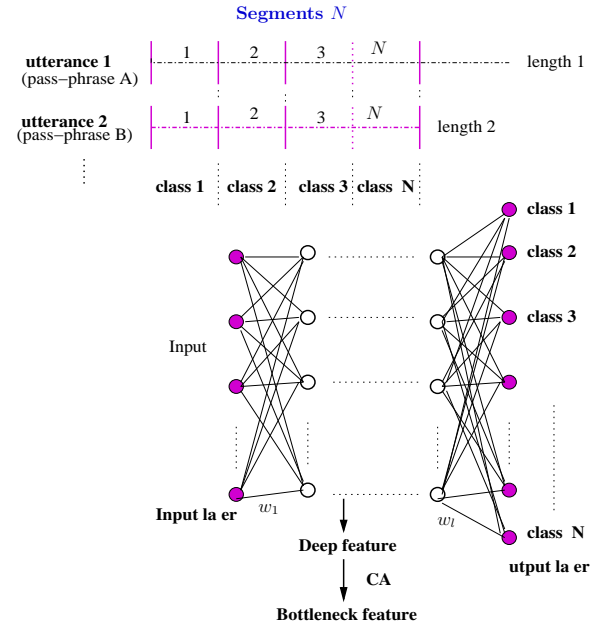


Figure 2: The unsupervised TCL based DNN feature extraction method for speech.

The outputs of the DNN, i.e. deep features, are projected onto a low dimensional space to get the BN feature for speaker verification. The number of output nodes in the last layer of

DNN is equal to the number of segments obtained by segmenting data into disjoint parts.

Utterances for speaker verification are often of different lengths primarily due to different text contents. In this work, the data set for the TCL training is constructed by forming a set of utterances of equal lengths by truncating the utterances longer than a predefined length and discarding a minority number of utterances shorter than the predefined length. Each utterance in the constructed data set are then uniformly segmented to segments of 6 frames each.

We also consider a variant of the above mentioned TCL training method, in which the parameters of the DNN are updated by pooling only the utterances of the same text content together at a time. We call it *TCL-seq*. The number of output nodes is kept the same as TCL for all different types of text utterances. The motivation of this is to see the impact of using homogeneous data for TCL training at each time, as compared to randomly mixing the training data in the aforementioned TCL method.

4.2. TCL transfer learning (TCL-tr)

In order to reduce the intra class variability within a segment of the same class during DNN training, in contrast to the TCL approach above, the TCL transfer learning method pools utterances of the same text together at a time and updates the DNN parameters as follows:

Step 1: Group the DNN training utterances based on their text contents, i.e. g_1, g_2, \dots regardless of speakers. Although this implementation groups utterances with same text together to leverage the transfer learning, the DNN training is still based on the unsupervised TCL concept.

Step 2: Read all utterances belonging to group g_1 , segment the utterances uniformly into a predefined number of segments n_1 regardless of lengths and train an initial DNN to discriminate the data among the different segments. The number of output nodes in the last layer of DNN is n_1 . Fig. 3 illustrates the segmentation of data and definition of classes in unsupervised manner for a general case, namely group g_i with n_i segments.

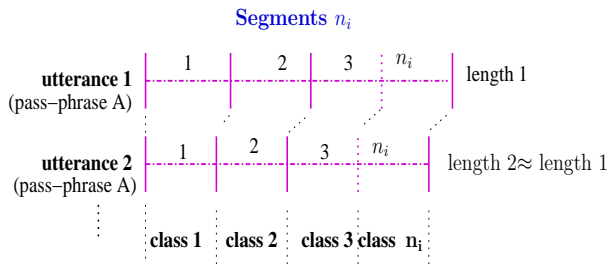


Figure 3: Segmentation of data and definition of classes in the sequential method

Step 3: Perform the DNN training for the utterances in group g_j with a predefined number of uniform disjoint segments n_j by using the model obtained by the previous step as an initial model (only modifying the number of output DNN nodes as per number of segments) as illustrated in Fig.4. Repeat this step until all of data groups are used for training.

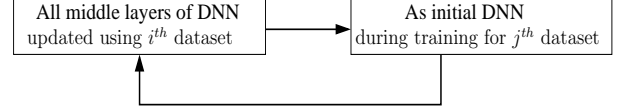


Figure 4: Sequential training

Step 4: Repeat training based on the transfer learning concept

Final DNN can be thought as representing the text independent model space as the model is trained using different contents of speech data in sequential manner. We consider different strategies to define the value of n (i.e. number of disjoint uniform segments) for different group of speech utterances. This gives several different features as follows:

TCL transfer learning with variable number of segments (TCL-tr-Var): DNN training data are grouped based on the pass-phrases and the number of segments for a group is calculated based on the number of phones available in the pass-phrase.

TCL transfer learning with variable number of segments + spkr+pass-phrase (TCL-tr-Var+spkr+phrase): This feature is similar to the *TCL-tr-Var*. Only difference is that additional speaker+pass-phrase discrimination training (as done in conventional DNNs) is performed, on top of the DNN model obtained through the *TCL-tr-Var*, by pooling all training data together. The number of output nodes in the last layer of DNN is equal to the number of speakers+pass-phrases. The motivation behind this is to see whether adding speaker+pass-phrases discrimination further improves the performance of the SV system as compared to *TCL-tr-Var*.

5. Experiment

Experiments are conducted on male speakers of the RSR2015 database (evaluation set of part1 i.e. *3sess-pwd_eval_m* task) [14] as per protocol. According to the protocol, there are 1708 speaker models to be trained. Three recording sessions are used to train each particular pass-phrase-wise target speaker model. The utterances are of very short duration on an average of 2-3s per speech signal. Test trials are divided into three types of non-target: target wrong, imposter correct and imposter wrong for system performance evaluation. Table 1 shows the number of trials available for the system evaluation on RSR2015.

Table 1: Number of trials for system evaluation on RSR2015

Database	No of trials in Non-target types		
	Target -wrong	Imposter -correct	Imposter -wrong
RSR2015	297076	573664	8318132

DNN setup and training: All DNNs are 7 layer feed-forward networks and are trained using the same learning rate and number of epoch. Each hidden layer consists of 1024 sigmoid units. The DNNs are trained using a 627 dimensional feature vector based on MFCC (57 dimensional) features with a context window of 11 frames (i.e. 5 frames left, current frame, 5 frames right). A cross-entropy objective function is utilized to discriminate the classes in the output layer of the DNN. Training data include 97 (50 male, 47 female) non-target speakers (disjoint from evaluation) from the development set of the RSR2015 database. Each speaker has a recording of 30 pass-phrases over 9 sessions. It gives approximately 26, 132 utterances for training the DNNs. In the conventional DNN method, the number

Table 2: Comparison of performances (% EER/MinDCF) of the TCL based unsupervised BN feature with baseline features for text-dependent speaker verification on 3sess-pwd_eval_m task of the RSR2015 database.

Non-target types	MFCC	Con. (spkr+phrase) DNN		TCL		TCL-seq		TCL-tr-Var		TCL-tr-Var+spkr+phrase	
		L2	L4	L2	L4	L2	L4	L2	L4	L2	L4
Target-wrong (TW)	0.87 /0.384	0.72 /0.321	0.57 /0.317	0.57 /0.306	0.57 0.291	0.66 /0.302	0.62 /0.300	0.69 /0.326	0.67 /0.305	0.69 /0.328	0.60 /0.291
Imposter-correct (IC)	1.16 /0.542	1.24 /0.569	1.17 /0.526	1.18 /0.535	1.08 0.507	1.19 /0.532	1.17 /0.520	1.22 /0.533	1.17 /0.545	1.26 /0.546	1.12 /0.512
Imposter-wrong (IW)	0.12 /0.051	0.10 /0.032	0.08 /0.027	0.09 /0.034	0.07 /0.029	0.07 /0.034	0.08 /0.032	0.09 /0.038	0.08 /0.031	0.10 /0.037	0.07 0.025
Average	0.71 /0.325	0.68 /0.307	0.60 0.290	0.61 /0.291	0.57 0.273	0.64 /0.289	0.62 0.284	0.66 /0.299	0.64 0.284	0.68 /0.304	0.60 0.276

of nodes in the output layer is equal to the number of speakers + pass-phrases in the training data, resulting in 127 nodes (97 speakers, 30 pass-phrases). The proposed DNN system also uses the same training data. In the TCL learning, the training utterance length is limited to 120 frames by discarding a minority number of utterances with length smaller than 120 frames and truncating utterances with length larger than 120 frames, so all training utterances are of 120 frames. The utterances are then segmented to 20 disjoint segments, considering that majority of the utterances have about 20 phones (varying from 14 to 30 phones). CNTK toolkit [17] is used for implementing the DNN and bottleneck feature extraction.

Bottleneck feature: Outputs from the second (L2) and fourth (L4) hidden layers are used as bottleneck features for this study as in [9] since they show the best performance. It gives 1024 dimensional deep features which are then projected onto a 57 dimensional vector space to align the dimension to the MFCC feature for a fair comparison. The deep features are normalized to zero mean and unit variance at utterance level before projecting onto the lower dimensional vector space using principle component analysis (PCA).

Gender-dependent GMM-UBM (512 mixtures, having diagonal covariance matrix) is trained using non-target speakers (438 male) data (4380 utterances) from TIMIT database [18]. The UBM training data are also used for training PCA. Speaker models are derived from the GMM-UBM with maximum a posteriori (MAP) adaptation using their respective training data. In test phase, test utterance $X = \{x_1, x_2, \dots, x_T\}$ is scored against the target specific model (obtained in training) λ_r and GMM-UBM λ_{ubm} . Finally, log likelihood ratio (LLR) value is calculated using the scores between the two models $LLR(X) = \frac{1}{T} \sum_{t=1}^T \{\log p(x_t|\lambda_r) - \log p(x_t|\lambda_{ubm})\}$. Three iterations (with value of relevance factor 10.0) are used in MAP. Only Gaussian mean vectors of the GMM-UBM are adapted.

For spectral analysis, 57 dimensional MFCCs (with RASTA [19] filtering) consisting of static C_1 - C_{19} cepstra, with Δ and $\Delta\Delta$ coefficients are extracted from speech signals using 10 ms frame shift and a 20 ms Hamming window. An energy based Voice Activity Detection (VAD) is applied to remove the less energized frames. Then, the energized feature vector are normalized to zero mean and unit variance at utterance level. System performance is evaluated in terms of equal error rate (EER) and minimum detection cost function (MinDCF) [20].

6. Results and discussion

We compare the performance of the TCL based unsupervised BN features with baseline cepstral and DNN based bottleneck features in Table 2, presented in term of EER (%) / MinDCF

($\times 100$) (2008 SRE cost function). From Table 2, it can be seen that DNN BN features using fourth layer (L4) show lower error rates as compared to their counterparts based on L2. All BN features (L4) give lower average error rates and MinDCF values (across different types of non-targets) as compared to cepstral (MFCC) feature as in [9].

For L2, most of the TCL BN features obtain slightly lower error rates as compared to the conventional DNN BN feature. In case of L4, the TCL BN features (*TCL*, *TCL-seq*) obtain better/very close TD-SV performance as compared to the conventional BN feature.

TCL and *TCL-seq* features show slightly lower error rates than the *TCL-tr-Var* feature. This indicates it is inefficient to initiate and train a new output layer for each set of utterances of the same sentence, i.e. the transfer learning. This can be database dependent and deserves further investigation. Furthermore, *TCL* shows better SV performance as compared to the *TCL-seq*. Incorporating speaker+pass-phrase discrimination in the *TCL-tr-Var* system further reduces the SV error rates as compared to that without.

Overall, the TCL unsupervised DNN BN feature is effective for text-dependent speaker verification. Moreover, the TCL method does not use any speaker/pass-phrase/phonetic specific label information and is able to take advantage of huge amount of unlabeled data available, which represents a huge advantage.

7. Conclusion

In this paper, we explored an unsupervised time-contrastive learning (TCL) method for training DNN based BN features for speaker verification, in which DNNs are trained to discriminate the temporal events across a speech signal. This is realized by uniformly segmenting the speech signal into a number of segments and assigning the same label to all speech frames in one segment but different labels to different segments. DNNs are then trained to discriminate data across the different time segments without any speaker or phonetic label in contrast to the conventional DNN BN feature extraction approaches which focus on discriminating speaker/pass-phrase/phone/both information in training using the labeled data. The output layer of the DNN is used for BN feature extraction. We consider different strategies for training the DNNs and yields different BN features. Experimental results on the RSR2015 database show that the TCL BN features outperform the conventional speaker and pass-phrase BN and cepstral features for TD-SV. This work confirms the feasibility of the TCL method for speech feature extraction. Besides its good performance, the TCL approach, as an unsupervised one, further has the big advantage of no need for labeled data.

8. References

- [1] T. Kinnunen and H. Li, "An Overview of Text-independent Speaker Recognition: from Features to Supervectors," *Speech Communication*, vol. 52, pp. 12–40, 2010.
- [2] Z.-H. Tan and B. Lindber, "Low-Complexity Variable Frame Rate Analysis for Speech Recognition and Voice Activity Detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 798–807, 2010.
- [3] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition," in *IEEE Signal Process. Mag.*, 2012, pp. 82–97.
- [4] M. McLaren, Y. Lei, and L. Ferrer, "Advances in Deep Neural Network Approaches to Speaker Recognition," in *Proc. of IEEE Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, 2015.
- [5] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep Neural Networks for Extracting Baum-welch Statistics for Speaker Recognition," in *Proc. of Odyssey Speaker and Language Recognition Workshop*, 2014, pp. 293–298.
- [6] E. Dahl, D. Yu, L. Deng, , and A. Acero, "Context-dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 30–42, 2012.
- [7] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep Neural Networks for Small Footprint Text-dependent Speaker Verification," in *Proc. of IEEE Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, 2014, pp. 4080–4084.
- [8] T. Fu, Y. Qian, Y. Liu, and Kai Yu, "Tandem Deep Features for Text-dependent Speaker Verification," in *Proc. of Interspeech*, 2014, pp. 1327–1331.
- [9] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep Feature for Text-dependent Speaker Verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.
- [10] C.-T. Do, C. Barras, V.-B. Le, and A. K. Sarkar, "Augmenting Short-term Cepstral Features with Long-term Discriminative Features for Speaker Verification of Telephone Data," in *Proc. of Interspeech*, 2013, pp. 2484–2488.
- [11] S. Ghahghajeh and R. Rose, "Deep Bottleneck Features for i-vector based Text-independent Speaker Verification," in *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 555–560.
- [12] A. Hyvarinen and H. Morioka, "Unsupervised Feature Extraction by Time-Contrastive Learning and Nonlinear ICA," in *Proc. of Neural Information Processing systems (NIPS)*, 2016.
- [13] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker Verification using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.
- [14] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent Speaker Verification: Classifiers, Databases and RSR2015," *Speech Communication*, vol. 60, pp. 56–77, 2014.
- [15] H. Delgado, M. Todisco, M. Sahidullah, A. K. Sarkar, N. Evans, T. Kinnunen, and Z.-H. Tan, "Further Optimisations of Constant Q Cepstral Processing for Integrated Utterance and Text-dependent Speaker Verification," in *Proc. of Spoken Language Technology Workshop (SLT)*, 2016.
- [16] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 19, pp. 788–798, 2011.
- [17] A. Agarwal et al., "An Introduction to Computational Networks and the Computational Network Toolkit," 2016.
- [18] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1," 1993, Web Download. Philadelphia: Linguistic Data Consortium.
- [19] H. Hermansky and N. Morgan, "Rasta Processing of Speech," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 578–589, 1994.
- [20] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The Det Curve in Assessment of Detection Task Performance," in *Proc. of Eur. Conf. Speech Commun. and Tech. (Eurospeech)*, 1997, pp. 1895–1898.

